公交路线的综合选择

摘要

本文针对以路线选择为核心的公交出行问题进行探讨。

对于问题一，在合理的模型假设基础上，考虑到实际中人们的出行心理，建立了以最小换乘次数为第一目标，以最小途径站数为第二目标的双目标规划模型，而乘车费用仅提供给查询者作为参考因素。在模型的求解中，并没有采用传统的图论方法，主要是考虑到，针对全局最优解的经典算法的计算量太大，对于这个接近 4000 个点的实际问题是不能接受的。而我们给出的“基于限定换乘次数的搜索算法”，在求解换乘 2 次之内路线时平均不到一分钟。至于换乘三次或三次以上的公汽站点，我们已通过程序考察出 92%以上的点与点之间是可以在换乘 2 次之内给出可行解的。在给出结论时，我们不是局限于“最佳路线”，而是从实际查询系统的人性化角度，对于不同的换乘次数，按照乘车时间由少到多，向查询者提供多条相对较优的乘车路线，并给出相应的乘车费用作为参考，以供其根据不同的需求自主选择对其而言最佳的乘车路线。

对于问题二，采用了将公交路线分类的方法，一是只乘坐公汽的路线，二是（一定）会乘坐地铁的路线。这样，问题二不仅延续了问题一的思想，而且巧妙地将“搜索算法”拓展为“分类搜索算法”。针对给定点间的多条乘车路线，我们又试图根据查询者的不同需求，对其优劣给出定量的评价标准，进而给出针对其需求的最佳乘车路线。首先，确定评价方案中的三个主要考虑因素,即换乘次数、时间、费用。在三种因素不冲突(即不存在此消彼长)的情况下，我们用贪心算法的思想要求方案尽量优化每一个因素。这样对于每对起点与终点可给出的方案数大致在 5 条以内，然后根据查询者需求，对评价方案的三个因素分配权值，最后利用层次分析法给定相对于每个查询者的最优方案，最终找出与之对应实际的出行路线。

问题三给出了允许步行对公交网络及对三个评价因素的影响，沿袭了前两问的思想，在问题 2 解的基础上，最有效的求得如何使用步行以及相应的路线选择方案。

模型扩展采用 Astar 算法，弥补了模型一局部最优解的缺点，验证了模型一求解过程中的结论。不论是求解的全面性，还是求解的时间性能都有很大程度提高，可作实际应用。并依照模型一问题三的思想尝试性给出的针对不同须求的最优解。

关键词　换乘次数 双目标 分类搜索 层次分析 Astar 算法

1 问题重述

第29届奥运会明年8月将在北京举行，公共交通工具（简称公交，包括公汽、地铁等）将会成为大部分人首选的出行工具。如今，城市的公交系统有了很大发展，北京市的公交线路已达800条以上。

与此同时，随着ATIS (先进的出行者信息系统) 的发展，需要设计一个公交查询系统，为出行者提供公交出行计划（核心为线路选择），以满足查询者的各种不同需求，使得公众的出行更加通畅、便利。我们将考虑以下问题：

(1) 仅考虑公汽线路，给出任意两公汽站点之间最佳线路选择；

(2) 同时考虑公汽与地铁线路，解决(1)；

(3) 假设知道所有站点之间的步行时间，给出任意两站点之间线路选择。

2 模型假设

(1) 题中所给的数据是真实可信的，即暂不考虑意外情况，如堵车现象等的发生。

(2) 对于环形路线，上行和下行分别对应于顺时针和逆时针线路。

(3) 分段计价中，“0～20 站：1 元” 表示在同一公汽线路上，途经站点数 $n \in (0,20], n \in N$ 时，票价为 1 元（即在相邻站点间行驶，途经站点数为 1）。

(4) 首次到达公交站点（即路线起始点）的时间是与当前问题无关的量，不影响路线选择，即不看作第一次转车而考虑转车时间。且起始站点是地铁站旁的公交车站时，不考虑从起始公交车站换乘到地铁的时间。

(5) 在问题 1 中，不考虑公交通过地铁站换乘情况。

(6) 对于环线问题, 当走到尽头时，公交上所有乘客必须下车，而地铁不必下。

3 系统符号

Li 公交线路编号（$i = 001,002 \ldots N_1$），$N_1 = 520$

Sj 公汽站点编号（$j = 0001,0002 \ldots N_2$），$N_2 = 3957$

$T1, T2$ 地铁线路编号

D_h 地铁站点编号（$h = 01,02 \ldots N_3$），$N_3 = 39$

t_1 相邻公汽站平均行驶时间(包括停站时间)，$t_1 = 3\min$

t_2 相邻地铁站平均行驶时间(包括停站时间)，$t_2 = 2.5\min$

$t_3, t_3^{(w)}$ 公汽换乘公汽平均耗时 $t_3 = 5\min$（其中步行时间 $t_3^{(w)} = 2\min$）

$t_4, t_4^{(w)}$ 地铁换乘地铁平均耗时 $t_4 = 4\min$（其中步行时间 $t_4^{(w)} = 2\min$）

$t_5, t_5^{(w)}$ 地铁换乘公汽平均耗时 $t_5 = 7\min$（其中步行时间 $t_5^{(w)} = 4\min$）

$t_6, t_6^{(w)}$ 公汽换乘地铁平均耗时 $t_6 = 6\min$（其中步行时间 $t_6^{(w)} = 4\min$）

q 地铁票价（无论地铁线路间是否换乘），$q = 3$ 元

S_o 查询中给定的起点

S_d 查询中给定的终点

（其它符号在首次使用时加以说明）

4 问题（1）模型的建立和求解

4.1 模型准备

4.1.1 公汽网络描述[1]

记$\Omega(S,L,R,A,B)$表示公汽网络，其中：

(1) 公汽站点集合

$$S=\{Sj \mid j=0001,0002\ldots N_2\}$$

(2) 公汽线路集合

对于公汽线路Li，是不考虑方向的。而事实上，Li包括上行和下行两个方向。

记l_k表示考虑上行和下行方向的公汽线路，其中$k=\begin{cases}2i-1 & l_k\text{表示上行线路}\\ 2i & l_k\text{表示下行线路}\end{cases}$（$i=001,002\ldots N_1$），则公汽线路集合表示为：

$$L=\{l_k \mid k=1,2\ldots 2N_1\}$$

(3) 线路-站点关系集合

$$R=\{r_k \mid k=1,2\ldots 2N_1\}$$

其中，r_k表示线路l_k上所有公汽站点按照行车线路所构成的有序序列，记为$r_k=<S_k{}^{(1)},S_k{}^{(2)},\cdots,S_k{}^{(m_k)}>$。该线路上共有$m_k$个站点，$S_k{}^{(1)}$和$S_k{}^{(m_k)}$分别是始发站和终点站，且$S_k{}^{(m)}(m=1,2,\cdots,m_k)$对应于$l_k$上第$m$个站点$Sj$。

(4) 站点-线路关系集合

$$A=\{a_j \mid j=1,2,\cdots,N_2\}$$

其中，$a_j=\{l_{j,1},l_{j,2},\cdots,l_{j,tj}\}$表示经过站点$Sj$的所有公汽线路（共$t_j$条）集合。

(5) 站点-线路-站点关系集合

$$B=\{b_j \mid j=1,2,\cdots,N_2\}$$

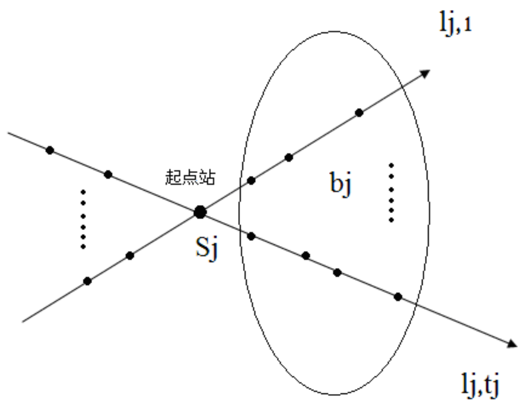

图1

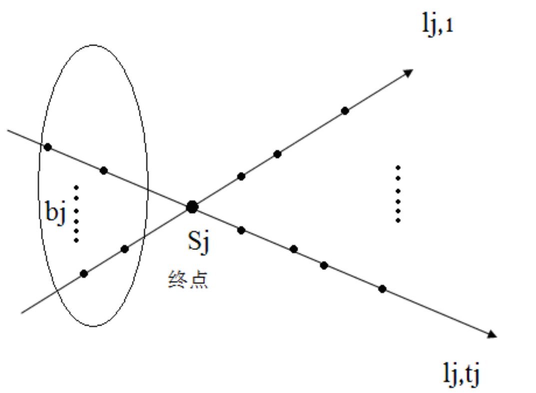

图1

其中，b_j表示经过站点Sj的所有公汽线路$l_{j,t}(t=1,2,\cdots,t_j)$所对应的有序序列$r_{j,t}$中，$Sj$前（后）所有站点的集合。

若Sj为起点站，b_j表示从Sj出发，可直达的所有站点的集合；若Sj为终点站，b_j表示所有可以直达Sj的站点的集合。即

$$b_j=\begin{cases}\bigcup_{k=1}^{tj}\bigcup_{p=m+1}^{m_k}S_k^{(p)} & Sj\text{为起点站(见图1左)}\\ \bigcup_{k=1}^{tj}\bigcup_{p=1}^{m-1}S_k^{(p)} & Sj\text{为终点站（见图1右）}\end{cases}$$

且 $\begin{array}{l} l_{j,1}\leftrightarrow r_{j,1}=<S_{j,1}^{(1)},\cdots,Sj,\cdots,S_{j,1}^{(m_{j,1})}> \\ \vdots \\ l_{j,tj}\leftrightarrow r_{j,tj}=<S_{j,tj}^{(1)},\cdots,Sj,\cdots,S_{j,tj}^{(m_{j,tj})}> \end{array}$ ，在每个有序序列 $r_{j,t}(t=1,2,\cdots,t_j)$ 中，Sj 的位置为 $m(j)$，均简记为 m 。

4.1.2 公汽票价函数

在考虑公汽线路方向的前提下，每条公汽线路要么采用单一票价，要么采用分段计价。记 $c(l_k)$ 表示票价性质函数，即 $c(l_k)=\begin{cases}1 & \text{线路}l_k\text{采用分段计价}\\ 0 & \text{线路}l_k\text{采用单一票价}\end{cases}$；记 $n_k(S_k^{(m1)},S_k^{(m2)})$ 表示在线路 l_k 上，从站点 $S_k^{(m1)}$ 到站点 $S_k^{(m2)}$ 的途经站点数（见模型假设(3)）。

则公汽票价函数 $f_k(S_k^{(m1)},S_k^{(m2)})$（单位：元）表示线路 l_k 上，从站点 $S_k^{(m1)}$ 到站点 $S_k^{(m2)}$ 的票价，即

$$f_k(S_k^{(m1)},S_k^{(m2)})=1 \quad c(l_k)=0$$

$$f_k(S_k^{(m1)},S_k^{(m2)})=\begin{cases}1 & 0<n_k(S_k^{(m1)},S_k^{(m2)})\le 20\\ 2 & 20<n_k(S_k^{(m1)},S_k^{(m2)})\le 40\\ 3 & n_k(S_k^{(m1)},S_k^{(m2)})>40\end{cases} \qquad c(l_k)=1$$

其中，$m1,m2=1,2,\cdots,m_k$。

4.2 模型建立

公交查询系统的核心问题，是对任意给定的起点和终点，给出最优乘车方案。

所谓“最优”，一般来说，在公交换乘的算法过程中，其实有多种优先策略的考虑，比如换乘次数最少、乘车时间最短、乘车费用最少等。每个人对公交换乘的路线选择其实都有所不同，因此乘车方案必须根据不同情况加以考虑。有些人需要考虑时间因素，有些人需要考虑费用因素。

考虑到实际中人们的出行心理——在乘车时间差距不大的情况下，宁可多坐几站，而换乘次数越少越好。研究也表明一般出行者以换乘次数为优先考虑的目标[2]，公交网络的设计以减少平均换乘次数为重要目标[3]。

而乘车时间，一方面，换乘次数越少，必然会使得换乘时间越小。但基于一般情况下,换乘次数不超过三次，即由于换乘次数而造成的乘车时间差最大也就是 15min（直达与换成三次）。这显然不是主要影响因素。另一方面，乘车时间取决于乘车线路的长短。由于相邻站点公汽的行驶时间相等，所以乘车过程中途经公汽站点数可以作为衡量乘车时间的一个主要标准。因此，在换乘次数一样的情况下，应该首选途径公汽站点最少的乘车路线，以减少乘车时间。

至于乘车费用，考虑到北京的整体消费水平，以及实际中人们在选择乘车路线时对乘车费用的态度，仅将其作为一个参考因素。考察对于同一起点和终点，在换乘次数与

乘车时间最小都相同时，判定方案优劣。不作为目标之一体现在算法中。

所谓乘车方案，是一个站点、线路的交替序列, 该序列说明从起点出发乘坐何线路，途中如何换乘, 直至到达终点。

因此，本文以最小换乘路线为第一目标，以最小途径站数为第二目标，以乘车费用为参考因素，建立双目标规划模型。

记从 S_o 到 S_d 的可行路径集合 $TR=\{TR_i \mid TR_i = S_o, l_{i1}, S_{i1}, l_{i2}, S_{i2}, \cdots, S_{i(D_i-1)}, l_{iD_i}, S_d\}$，其中，$TR_i$ 表示从 S_o 出发，乘坐 l_{i1} 到 S_{i1}，在 S_{i1} 换乘 l_{i2} 到 S_{i2}，…，在 $S_{i(Di-1)}$ 换乘 l_{iDi} 直至 S_d。该路径中，

(1) 第一目标：换乘次数 D_i

(2) 第二目标：途径站数 U_i

$$U_i = n_{i1}(S_o, S_{i1}) + n_{i2}(S_{i1}, S_{i2}) + \cdots + n_{iDi}(S_{i(Di-1)}, S_d) = \sum_{d=1}^{Di} n_{id}$$

(3) 乘车时间：

$$Time(S_o, S_d)_i = t_1 * U_i + t_3 * D_i$$

(4) 乘车费用：

$$\begin{aligned} Cost(S_o, S_d)_i &= f_{li1}(S_o, S_{i1}) + f_{li2}(S_{i1}, S_{i2}) + \cdots + f_{liDi}(S_{i(Di-1)}, S_d) \\ &= \sum_{d=1}^{Di} f_{li\,d} \end{aligned}$$

(5) 出行者的目标函数[1]：

$$\underset{TR_I}{Max} \quad W(D_i, U_i)$$

$$s.t. \quad \begin{cases} \dfrac{\partial W}{\partial D_i} < 0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (1) \\ \dfrac{\partial W}{\partial U_i} < 0 \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (2) \\ \left|\dfrac{\partial W}{\partial D_i}\right| \ \rangle\ \rangle \ \left|\dfrac{\partial W}{\partial U_i}\right| \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (3) \end{cases}$$

其中，(1)、(2)表示目标极小化约束；(3)表示 D_i 为第一目标，U_i 为第二目标。

4.3 模型求解与结果分析

在实际查询系统中，考虑到“最优乘车方案”并不能满足查询者各种不同的需求。因此，对于给定的起点 S_o 和终点 S_d，**首先**根据模型假设(1), 将第一目标分解为直达、换乘一次、换乘两次和换乘三次 4 种情况；**再**针对每种情况，以第二目标搜索途经站点最少的乘车方案；**最后**，综合考虑换乘次数和乘车时间，提供多条“较优乘车路线”，以及相应的乘车费用，使得查询者可以根据各自的需求自主选择。

4.3.1 搜索算法

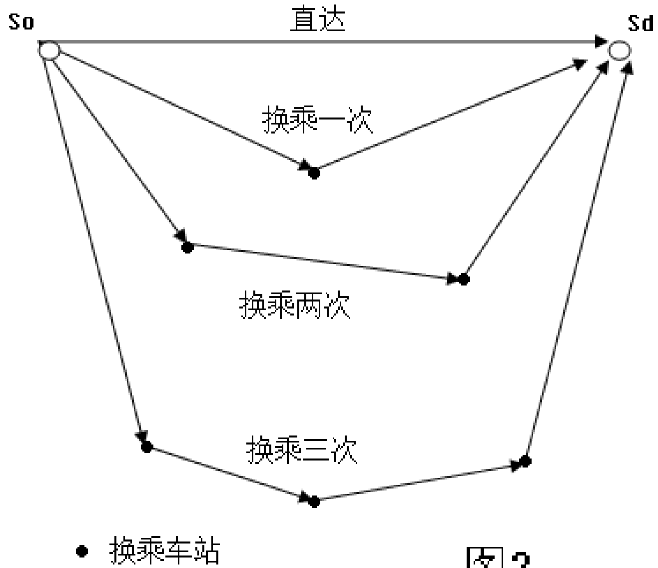

图2

*step*1 输入起点 S_o 和终点 S_d。

*step*2 搜索直达线路：遍历每一条公交线路，搜索出可以由起点 S_o 不经换乘直达终点 S_d 的线路。选出途经车站数最少的三条线路，并记录对应的时间及其中最少的时间 $t_{\min}(0)$ 做为换乘 1 次的时间约束。

*step*3 搜索换乘 1 次的线路：找出由 S_o 出发可以直达的站点的集合 A_o，以及可以直达 S_d 的站点的集合 A_d；

若 $A_o \cap A_d \neq \Phi$，则在交集中的任意一点换乘 1 次即可以从 S_o 到达 S_d，求出换乘 1 次的路线，记录对应的时间并与 $t_{\min}(0)$ 比较，当该时间小于 $t_{\min}(0)$ 时，保留；同 *step*2 中一样，选出最少的时间 $t_{\min}(1)$ 做为换乘 2 次的时间约束。

若 $A_o \cap A_d = \Phi$，则表示从 S_o 到 S_d 没有经过 1 次换乘可到达的线路，记 $t_{\min}(1) = \infty$。

*step*4 搜索换乘 2 次的线路：找出由 S_o 出发可以直达的站点的集合 A_o，将 A_o 中每一个点做为起点转 *step*3，求出从该点换乘 1 次的线路，从而得到从 S_o 到 S_d 经过 2 次换乘可到达的线路；记录对应的时间并与 $t_{\min}(1)$ 比较，当该时间小于 $t_{\min}(1)$ 时，保留；选出最少的时间 $t_{\min}(2)$ 做为换乘 3 次的时间约束，若没有换乘 2 次的线路，则记 $t_{\min}(2) = \infty$。

*step*5 搜索换乘 3 次的线路：找出由 S_o 出发可以直达的站点的集合 A_o，将 A_o 中每一个点做为起点转 *step*4，求出从该点换乘 2 次的线路，从而得到从 S_o 到 S_d 经过 3 次换乘可到达的线路；记录对应的时间并与 $t_{\min}(2)$ 比较，当该时间小于 $t_{\min}(2)$ 时，保留。

4.3.2 结果分析（评价说明以(1)为例）

(1) S3359→S1828

	S_o	l_{i1}	S_{i1}	l_{i2}	S_{i2}	l_{i3}	S_d	Time (min)	Cost (元)
直达	3359	无直达车					1828	--	--
换乘一次	3359	872	1784	167; 217	--		1828	101	3

<table>
<tr><td rowspan="4">换乘两次</td><td rowspan="4">3359</td><td>30，245，264，704，731，871 948</td><td>2903</td><td>970</td><td>1784</td><td>334，434</td><td rowspan="4">1828</td><td rowspan="4">64</td><td rowspan="4">3</td></tr>
<tr><td>968</td><td>2027，1746，3697，3727</td><td>970</td><td>1784</td><td>334，434</td></tr>
<tr><td>648</td><td>2027，1746</td><td>970</td><td>1784</td><td>334，434</td></tr>
<tr><td>938</td><td>2027</td><td>970</td><td>1784</td><td>334，434</td></tr>
<tr><td>换乘三次</td><td>3359</td><td colspan="5">无满足算法要求的换乘三次公汽线路（即换乘三次公汽线路所对应的乘车时间均大于 64min）</td><td>1828</td><td>--</td><td>--</td></tr>
</table>

经分析上表，可知：

a) 从站点 S3359 到站点 S1828，无直达公汽线路。
b) 换乘三次的可行路径所对应的乘车时间均大于 64min（换乘两次的最小乘车时间）。这样，在不考虑乘车费用的情况下，忽略换乘三次的可行路径。
c) 乘车费用几乎一样，这一点验证了“在考虑最优乘车路线时，可以忽略费用问题”。
d) 存在并行路线，如从 S1784 到 S1828，334（L167 下行）与 434（L217 下行）均只有一站路。
e) 换乘两次的乘车时间（64 min）远小于换乘一次的乘车时间（最少 101min），这时，就需要查询者根据各自的需求选择相应的路线。
f) 我们已通过程序 bus 考察出 92%以上的点与点之间是可以在换乘 2 次之内给出可行解，这就验证了限定换乘次数的合理性。
g) 在模型建立中，我们以最小乘车次数为第一（重要）目标，以最小途经站数为第二（次要）目标，这样可能会造成换乘次数很少，途经站数很多，使得乘车时间随着换乘次数的增加而急剧减少，正如 e)。

    但我们所给出的“搜索算法”却巧妙地解决了该问题，由于当前仅考虑换乘次数和总时间，查询者只在二者之间取最看重的，因此可以列出表格，把选择权留给了查询者。即把第一目标分解为直达、换乘一次、换乘两次和换乘三次，分别计算每种情况下的可行路径，并且保证乘车时间随着换乘次数的增多而不增加。例如本例中，换乘一次的最少乘车时间为 101min，在计算换乘两次的公汽线路时，当乘车时间大于该最小值（101min）时，不考虑该可行路径。

(2) S1557→S0481（无直达、换乘一次）

换乘	S_o	l_{i1}	S_{i1}	l_{i2}	S_{i2}	l_{i3}	S_{i3}	l_{i4}	S_d	Time（min）	Cost（元）

两次	1557	168，726	1919	378	3186	920	--	--	481	106	3
三次	1557	168，726	1919	378	3186	181	903	478,507,624,893,920,1027,1031	481	99	4

(3) S0971→S0485（无直达）

换乘	S_o	l_{i1}	S_{i1}	l_{i2}	S_{i2}	l_{i3}	S_{i3}	l_{i4}	S_d	Time（min）	Cost（元）
一次	971	26	2184	834	--	--	--	--	485	128	3
两次	971	26	2517	580	2159	937	--	--	485	103	3
三次	971	25,48,187,238,526	1609	895	2113	4	2654,1729,3766,1691,1383,1381,2019,2017	937	485	105	4

(4) S0008→S0073（无直达）

换乘	S_o	l_{i1}	S_{i1}	l_{i2}	S_{i2}	l_{i3}	S_{i3}	l_{i4}	S_d	Time（min）	Cost（元）
一次	8	318	2683,291,3614,491,2559,3315	116	--	--	--	--	73	83	2
			400,2633,3053	947							
两次	8	86,395,926	1383	580	2184	689	--	--	73	67	3
三次	8	395	1383,1691,3766	3,564,580	1327	655	525	205	73	63	4

(5) S0148→S0485（无直达、换乘一次、换乘两次）

换乘	S_o	l_{i1}	S_{i1}	l_{i2}	S_{i2}	l_{i3}	S_d	Time（min）	Cost（元）
两次	148	615	36	311	2210,3332,3351	834	485	106	3
三次	148	615	3604	162,246,708	2210,3332,3351	834	485	102	4

(6) S0087→S3676（无直达、换乘三次）

换乘	S_o	l_{i1}	S_{i1}	l_{i2}	S_{i2}	l_{i3}	S_d	Time（min）	Cost（元）
一次	87	907	3496	418	--	--	3676	65	2
两次	87	42,411,908	88	461	427	193, 924	3676	46	3

(以上均为部分较优乘车方案，详见附件)

5 问题二 同时考虑公汽与地铁路线

5.1 模型分析

从直观上讲，乘坐地铁与乘坐公汽相比，一般乘车时间相对较少，而乘车费用相对较高。当考虑地铁线路时，由于交通工具间的换乘时间、相邻站点的行驶时间、线路和票价之间的差异性，使得按照问题(1)的思路，利用“搜索算法”寻找最优路线，是困难的。

而且，在问题(1)的求解结果中，从起点S_o到终点S_d，公汽乘车路线不止一条。我们分别对于不同的换乘次数，按照乘车时间由少到多，向查询者提供多条相对较优的乘车路线，并给出相应的乘车费用作为参考，以供其根据不同的需求自主选择对其而言最佳的乘车路线。这很好地符合了实际查询系统的人性化要求。

但对于从起点S_o到终点S_d的多条乘车路线，问题（1）中，我们并没有根据查询者的不同需求，对其优劣给出定量的评价标准，进而给出针对其需求的最佳乘车路线。

事实上，从公汽站点S_o（起点）到S_d（终点），乘车路线中要么会乘坐地铁，要么只乘坐公汽。这样，我们可以首先利用问题(1)，找出只乘坐公汽的一些较优路径；再考虑（一定）会乘坐地铁的乘车方案中的一些较优路径；最后，针对出行考虑的不同因素，给每种因素按照查询者的不同需求分配相应的权值，利用层次分析法，得出最佳路线。

5.2 模型准备

5.2.1 地铁站点间‘最短行驶时间’矩阵

地铁只有两条线路$T1$,$T2$,共39个站点，分别为$D01, D02, \cdots, D39$，其中$T2$为环线，$D12, D18$为地铁换乘站。对于任意的两个地铁站点D_i, D_j，乘坐地铁从D_i到D_j的最短时间唯一确定，记$e(D_i, D_j)$或e_{ij}，且$e_{ij}=0(i=j)$。

则$E_{39\times39}=(e_{ij})_{39\times39}$表示地铁站点之间的最短时间矩阵，且为对称阵($e_{ij}=e_{ji}$)，对角线上的元素全为0。

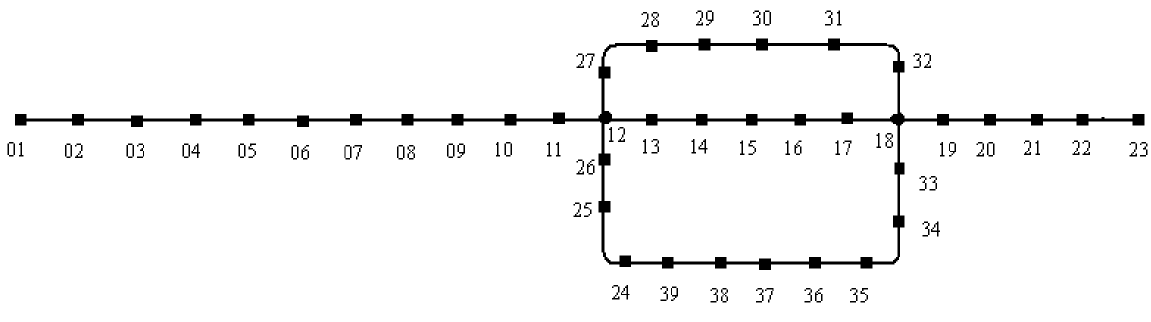

图3 地铁线路示意图

求解算法如下：

1) 线路$T1$（有向）上的地铁站点集合$TT1=\{1,2,\cdots,23\}$；

   环形线路$T2$（有向）上的地铁站点集合$TT2=\{12,18,24,25,\cdots,39\}$；

   当$i, j \in T_2$，u_{ij}表示按顺时针方向，D_i和D_j之间的站数（当D_i、D_j相邻时，u_{ij}=1)。

2) For i=1 to 39

   For j=(i+1) to 39

   分别判断 i,j 属于$TT1$还是$TT2$；

①若$i,j\in T_1$，则$e_{ij}=(j-i)*t_2$；

②若$i,j\in T_2$，则$e_{ij}=Min\{u_{ij},18-u_{ij}\}*t_2$；

③若$i\in T1,j\in T2$，则

$$e_{ij}=Min\{[|12-i|+Min\{u_{12,j},18-u_{12,j}\}],[|18-i|+Min\{u_{18,j},18-u_{18,j}\}]\}*t_2+t_4;$$

④若$i\in T2,j\in T1$，则$e_{ij}=e_{ji}$。

End

End

5.2.2 地铁站点间的‘换乘’矩阵

对任意两个地铁站点D_i,D_j，唯一确定从D_i到D_j的最短时间e_{ij}，而e_{ij}又对应于一条以D_i为起点、以D_j为终点的地铁路径，该路径可标记出从D_i到D_j是否需要换乘地铁，并记$v_{ij}=\begin{cases}1 & \text{从}D_i\text{到}D_j\text{需要换乘地铁}\\ 0 & \text{从}D_i\text{到}D_j\text{不需要换乘地铁}\end{cases}$。

则$V_{39\times39}=(v_{ij})_{39\times39}$表示地铁站点之间的‘换乘’矩阵，且为对称阵，对角线上元素全为 0。

5.3 模型建立

同时考虑公汽线路和地铁线路的情况下，从起点S_o到终点S_d，存在不止一条公交路线。但这些公交路线总是可以分为两类：一是只乘坐公汽的路线，记为Ⅰ；二是（一定）会乘坐地铁的路线，记为Ⅱ。

对于给定S_o和S_d，每类中的每条公交路线，均有一个换乘次数η（包括同类和不同类公交之间的换乘）、乘公交时间$Time$和乘公交费用$Cost$。对出行者而言，$(\eta,Time,Cost)$均相同的公交路线是无差别的。

根据模型分析，采用分类搜索算法，并建立基于层次分析法的最佳路线选择模型。具体步骤如下：（给定S_o和S_d）

（1）第Ⅰ类搜索

① 利用问题(1)中的搜索算法，分别找出从S_o换乘η次到达S_d（$\eta=0,1,2,3$）的路线集合$LR(\mathrm{I})^{(\eta)}$。若不存在（或是因为S_o与S_d间不存在换乘η次的可行路径，或是因为乘车时间过长（见搜索算法）而被算法筛去），以ϕ表示。

② 在集合$LR(\mathrm{I})^{(\eta)}$中，将$(Time(\mathrm{I})_k^\eta,Cost(\mathrm{I})_k^\eta)$相同的公交路线记为$lr(\mathrm{I})_k{}^{(\eta)}$。则其唯一对应$(\eta,Time(\mathrm{I})_k^\eta,Cost(\mathrm{I})_k^\eta)$，表示从$S_o$出发可通过集合$lr(\mathrm{I})_k{}^{(\eta)}$中任意公交路线换乘$\eta$次到达到$S_d$，并且乘公交时间为$Time(\mathrm{I})_k^\eta$，乘公交费用为$Cost(\mathrm{I})_k^\eta$。

显然，$LR(\mathrm{I})^{(\eta)}=\bigcup_k lr(\mathrm{I})_k{}^{(\eta)}$。

③ 以$Min\{Time(\mathrm{I})_k^\eta\}$筛选$lr(\mathrm{I})^{(\eta)}$，再从结果中找出最小乘公交费用$Cost(\mathrm{I})_{k*}^\eta$，并记对应的公交路线集合为：$lr(\mathrm{I})_{k*}{}^{(\eta)}\leftrightarrow(\eta,Min\{Time(\mathrm{I})_k^\eta\},Cost(\mathrm{I})_{k*}^\eta)$。

④ 以$Min\{Cost(\mathrm{I})_k^\eta\}$筛选$lr(\mathrm{I})^{(\eta)}$，再从结果中找出最小乘公交时间$Time(\mathrm{I})_{k**}^\eta$，并记对应的公交路线集合为：$lr(\mathrm{I})_{k**}{}^{(\eta)}\leftrightarrow(\eta,Time(\mathrm{I})_{k**}^\eta,Min\{Cost(\mathrm{I})_k^\eta\})$。

⑤ 第Ⅰ类搜索结果：

$$\left.\begin{aligned}&lr(\mathrm{I})_{k^*}{}^{(\eta)} \leftrightarrow (\eta, Min\{Time(\mathrm{I})_k^\eta\}, Cost(\mathrm{I})_{k^*}^\eta)\\&lr(\mathrm{I})_{k^{**}}{}^{(\eta)} \leftrightarrow (\eta, Time(\mathrm{I})_{k^{**}}^\eta, Min\{Cost(\mathrm{I})_k^\eta\})\end{aligned}\right.\quad（\eta = 0,1,2,3）$$

(2) 第II类搜索

由于同时考虑公汽和地铁线路，则无法利用问题(1)中的“搜索算法”找出 $LR(\mathrm{II})^{(\eta)}$。

记集合 $D_h = \{S_1, S_2, \cdots, S_{nh}\}(h = 1,2,\cdots,39)$ 表示地铁站点 D_h 处，地铁−公汽换乘站点集合。

记 $\Phi = \bigcup_{h=1}^{39} D_h$，表示所有地铁−公汽换乘站点集合。

根据起点 S_o 和终点 S_d 是否可换乘地铁，分为以下四种情况：

$$\begin{cases}1) \quad S_o, S_d \in \Phi\\2) \quad S_o \in \Phi, S_d \notin \Phi\\3) \quad S_o \notin \Phi, S_d \in \Phi\\4) \quad S_o, S_d \notin \Phi\end{cases}$$

1) $S_o, S_d \in \Phi$

S_o, S_d 均为地铁−公汽换乘站点，不妨设各自对应的地铁站为 D_i, D_j。

① 从地铁站点间‘最短行驶时间’矩阵 $E_{39\times39}$ 中调出 e_{ij}；

② 从地铁站点间的‘换乘’矩阵 $V_{39\times39}$ 中调出 v_{ij}；

③ 第II类搜索结果：$lr(\mathrm{II})_{k^*}{}^{(\eta)} \leftrightarrow (v_{ij}, e_{ij}, q)$

2) $S_o \in \Phi, S_d \notin \Phi$

仅 S_o 为地铁−公汽换乘站点，不妨设其对应的地铁站为 D_i，而离开地铁站的换乘站点是 $D_j = \{S_1, S_2, \cdots, S_{nj}\}$，选择 $S_k(k = 1,2,\cdots,nj)$ 换乘公汽。

我们仍然延续问题(1)中以“换乘次数最少”为第一目标的思想。

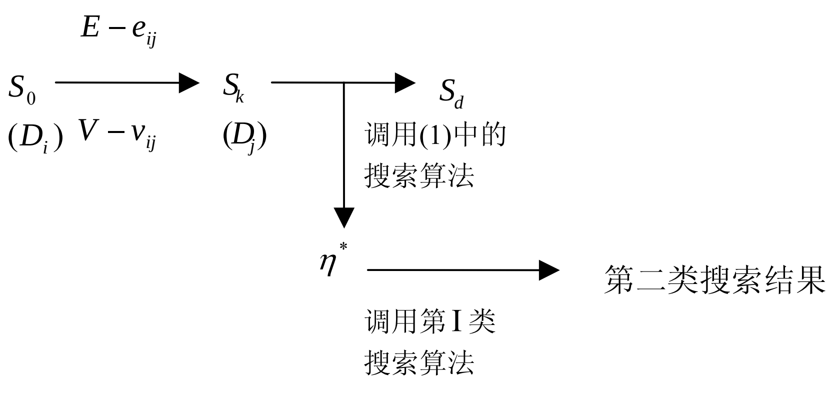

图 3 算法示意图

① 从地铁站点间‘最短行驶时间’矩阵 $E_{39\times39}$ 中调出 e_{ij}；

② 从地铁站点间的‘换乘’矩阵 $V_{39\times39}$ 中调出 v_{ij}；

③ 从 S_k 换乘公汽去 S_d 时，调用问题(1)中的搜索算法，得到最少的换乘次数 η^*。

④ 在确定η^*后，调用第I类搜索算法，得到$\begin{array}{l} lr(\mathrm{I})_{k*}{}^{(\eta^*)} \leftrightarrow (\eta^*, Time(\mathrm{I})_{k*}^{\eta^*}, Cost(\mathrm{I})_{k*}^{\eta^*}) \\ lr(\mathrm{I})_{k**}{}^{(\eta^*)} \leftrightarrow (\eta^*, Time(\mathrm{I})_{k**}^{\eta^*}, Cost(\mathrm{I})_{k**}^{\eta^*}) \end{array}$；

⑤ 第二类搜索结果：

$$lr(\mathrm{II})_{k*}{}^{(\eta^*)} \leftrightarrow ((v_{ij}+1+\eta^*),(e_{ij}+t_5+Time(\mathrm{I})_{k*}^{\eta^*}),(q+Cost(\mathrm{I})_{k*}^{\eta^*}))$$
$$lr(\mathrm{II})_{k**}{}^{(\eta^*)} \leftrightarrow ((v_{ij}+1+\eta^*),(e_{ij}+t_5+Time(\mathrm{I})_{k**}^{\eta^*}),(q+Cost(\mathrm{I})_{k**}^{\eta^*}))$$
$$s.t. \qquad \underset{Dj,Sk}{Min}\{v_{ij}+1+\eta^*\}$$

3) $S_o \notin \Phi, S_d \in \Phi$

类似于 2)，略

4) $S_o, S_d \notin \Phi$

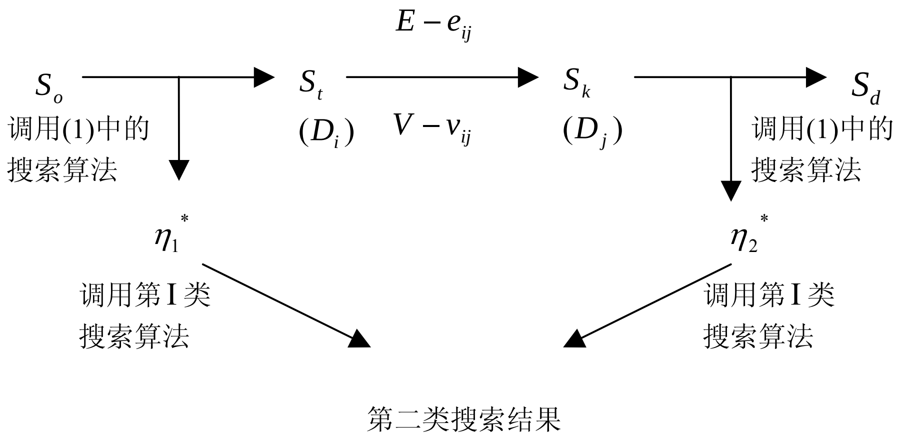

图 4

统一的算法描述如下：

*step*1 输入起点S_o和终点S_d。

*step*2 判断起点S_o和终点S_d是否在地铁附近（即可以直接乘坐地铁）。若两者都在地铁附近，则转*step*3；若S_o在地铁附近，S_d不在，则转*step*4；若S_d在地铁附近，S_o不在，则转*step*5；若两者都不在地铁附近，则转*step*6。

*step*3 求S_o和S_d之间乘坐地铁的最短路，结束。

*step*4 寻找从每一地铁站出发可直达S_d的公交线路，对应每一条线路求相应地铁站经过无换乘或 1 次换乘地铁到S_o的最短路，即可得到 S_o–地铁–公汽–S_d 的线路，在所有线路中选出时间最少的做为最优解，结束。

*step*5 寻找从S_o出发可直达每一地铁站的公交线路，对应每一条线路求相应地铁站经过无换乘或 1 次换乘地铁到S_d的最短路，即可得到 S_o–公汽–地铁–S_d 的线路，在所有线路中选出时间最少的做为最优解，结束。

*step*6 寻找从S_o出发可直达每一地铁站的站点–线路–站点关系集合B_o，以及从每一地铁站出发可直达S_d的站点–线路–站点关系B_d，求出从S_o对应地铁站到S_d对应地铁站的最短路，即可得到S_o–公汽–地铁–公汽–S_d，在所有线路中选出时间最少的做为最优解，结束。

(3) 基于层次分析法的最佳路线选择

第I类搜索和第II类搜索给出的结果，均是形如“$lr^{(\eta)} \leftrightarrow (\eta, Time, Cost)$”的从起点至终点的可能最佳公交路线，可视为方案层。每条路线有唯一确定的换乘次数、乘公交时间和乘公交费用，而这三个因素正是人们出行所要考虑的，可看作准则层。由于个人的不同需求，对三个因素的侧重会不同。如上班族侧重时间，生活条件差的人更关注费用，而根据大多数人的出行心理，换乘次数越少越好。而目标则是乘公交从起点到终点。于是，便可以采用层次分析法，根据个人需求，选择最佳路线。

①建立层次结构模型

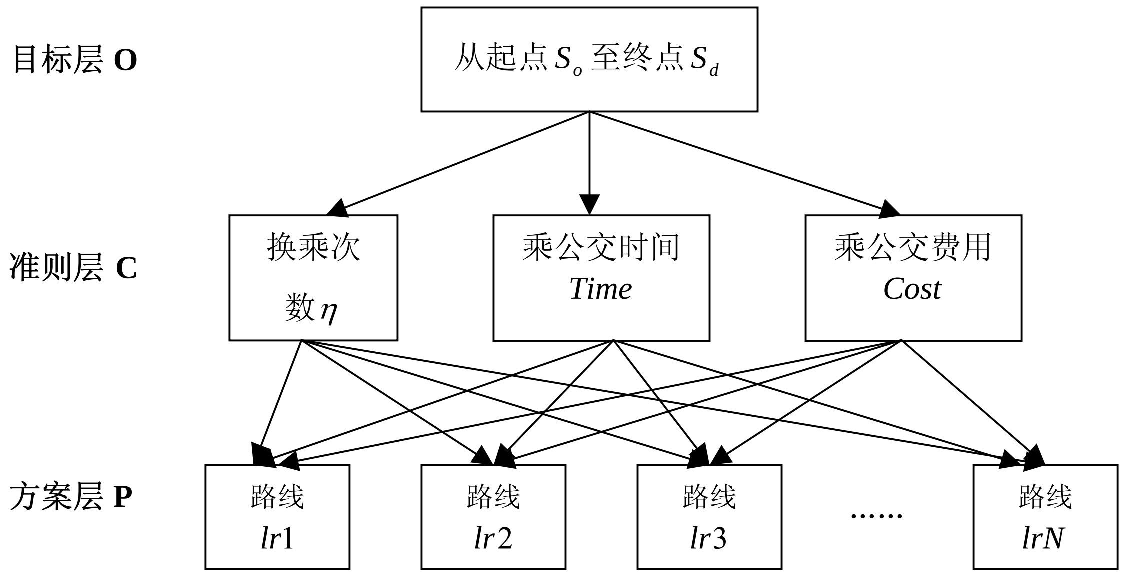

图 5 层次分析示意

②确定准则层（C）对目标层（O）的权重 W_0

查询者根据不同的需求，对于三个准则——换乘次数、乘公交时间和乘公交费用，会有不同的侧重程度。因此，可以根据查询者的需求，对各准则采用分配权值的方法来确定。

不妨设给换乘次数、时间和费用分配的权值分别是 α, β, γ。则可构造成对比较阵

$A = (a_{ij})_{3\times 3} = \begin{pmatrix} 1 & \frac{\alpha}{\beta} & \frac{\alpha}{\gamma} \\ \frac{\beta}{\alpha} & 1 & \frac{\beta}{\gamma} \\ \frac{\gamma}{\alpha} & \frac{\gamma}{\beta} & 1 \end{pmatrix}$。A 为一致阵，$W_0 = \{\alpha\ \beta\ \gamma\}^T$， $CI = 0, CR = 0$。

③确定方案层（P）对准则层（C）的权重 W_1

已知 $lr_i = (\eta_i, Time_i, Cost_i)$；

不妨设方案层对第 k 个准则的成对比较阵为 $B_k = (b_{ij}{}^{(k)})_{N\times N} (k = 1,2,3)$，其中，

$b_{ij}{}^{(k)} = \dfrac{lr_i(k)}{lr_j(k)}$ $\quad lr_i(1) = \eta_i, lr_i(2) = Time_i, lr_i(3) = Cost_i$。

由 B_k 计算 $w_{k3\times1}$，则 $W_1=[w_1,w_2,w_3]$。

④计算组合权向量

5.4 模型求解

（仅以 $\alpha=0.5,\beta=0.3,\gamma=0.2$ 为例）

(1) S3359→S1828

换乘次数	时间	费用	权值
1	101	3	**0．2910**
2	64	3	0．3410
3	51	5	0．3679

(2) S1557→S0481

换乘次数	时间	费用	权值
2	106	3	**0．2945**
3	99	4	0．3758
2	108	5	0．3297

(3) S0971→S0485

换乘次数	时间	费用	权值
1	128	3	**0.1977**
2	103	3	0.2416
3	105	4	0.3189
2	67.5	5	0.2419

(4) S0008→S0073

换乘次数	时间	费用	权值
1	83	2	**0.2466**
2	67	3	0.3382
3	53.5	5	0.4152

(5) S0148→S0485

换乘次数	时间	费用	权值
2	106	3	**0.3164**
3	102	4	0.3591
2	59	5	0.3245

(6) S0087→S3676

换乘次数	时间	费用	权值
1	65	2	0.3631
2	46	3	0.4929
0	25	3	**0.1440**

综上，可得个站点间最佳路线如下：

$S3359 \xrightarrow{L872} S1784 \xrightarrow{L334} S1828$ 时间 110 费用 3

$S1557 \xrightarrow{L168} S1919 \xrightarrow{L378} S3186 \xrightarrow{L920} S0481$ 时间 106 费用 3
$S0971 \xrightarrow{L25} S1609 \xrightarrow{L895} S2113 \xrightarrow{L4} S2654 \xrightarrow{L937} S0485$ 时间 105 费用 4
$S0008 \xrightarrow{L318} S0400 \xrightarrow{L957} S0073$ 时间 83 费用 2
$S0148 \xrightarrow{L615} S0036 \xrightarrow{L311} S2210 \xrightarrow{L834} S0485$ 时间 106 费用 3
$S0087 \rightarrow D27 \xrightarrow{T2} D36 \rightarrow S3676$ 时间 25 费用 3

6 问题三

6.1 模型分析

问题三假设知道所有站点之间的步行时间，在问题二的基础上给出任意两站点之间线路选择问题的数学模型。 我们遵从整个论文假设前提，即附录 1 的基本参数设定，假设任意相邻两个相邻的公交站点间步行平均时间为 t。依据常识，我们只考虑汽车站点间可步行，地铁的路线不能采用步行。

与问题一发展到问题二的情况一样，增加了步行方式，同样是增加了新的可能的路径。在前两问中，公交路线都是有向的，即在站点 Sj 只能在 b_j 中选择。但步行不需要考虑此约束，即可以沿当前的路线逆行。设 cj 表示在有序序列 $r_{j,t}$ 中，b_j 和 Sj 的补集。见图 3 如站点 Sj 为路线选择的起始点时，考虑步行，即可以选择 cj 中的站点作为下一站点。

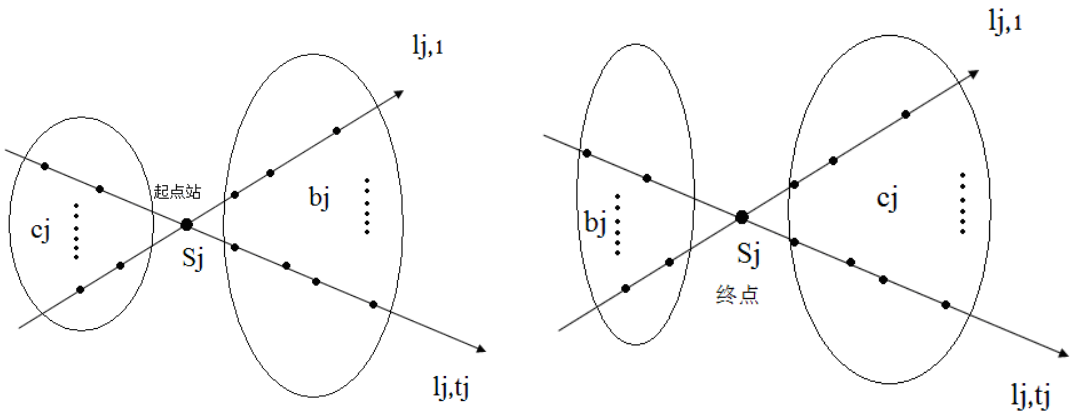

图3 图3

那么，出行路线的选择问题便与四个因素，即换乘次数、总时间、花费、步行次数 n 有关。

下面分析步行次数对前三种因素的影响，在此只列举一些与之前相比可能优化的情形。

如图 4：$Ai \sim Bj$ 有 1 或很少的几站，站点 A 乘公交到站点 Ai 后换乘一次到达 Bj 后，再换乘一次最终到达 B。换乘次数为 2，步行次数为 0。可以选择从 Ai 走到 Bj，减少换乘一次，同时减少花费，但增加步行次数，可能增加时间。 比如，从 S3359 到 S1828 的方案中，出现的从中转站 S1784 到 S1828 只有一站的情况就涵盖在上述情况里。

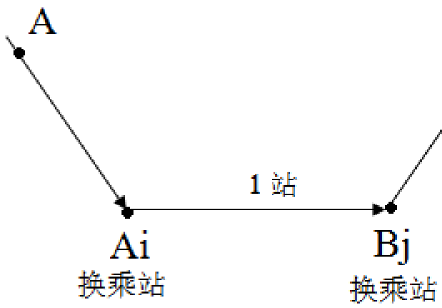

图4

还有可能的情况存在于站点 A 到达站点 B 刚好 21/31/41 站，如果该路线是分段计价的情况，那么如果步行最后的一站，会节省开销。

通过上述的分析，我们发现考虑步行的模型存在的大致好处有：

1）增加新的可能的路径，因为步行不考虑公交车站的方向性，即让站点 Sj 选择 cj 中的站点；

2）节省费用，合理的步行安排可以降低费用或减少换乘次数，但可能以增加总时间和步行次数为代价。

6.2 模型建立与算法

基于模型分析，我们应尽量合理安排何时使用步行。

为利用优点 1），我们在仅考虑步行时，将原来公交网的有向图变为无向图。（在考虑公汽时的图依然是有向的）。

为利用优点 2），我们在问题 2 给定的优良解基础上再次改良。对问题 2 的解处理如下：

规定步行的车站数不超过 5 站，然后从只步行 1 站开始，分别测试步行 1 站、2 站、3 站、4 站、5 站。每次测试对四个评价因素，换乘次数、总时间、花费、步行次数进行判断，得到对该次测试线路上安排步行的最佳路段，然后返回解，即换乘次数、总时间、花费、步行次数的值，记录这 5 次测试的值，与在与问题 2 的解进行比较，比较的方法依然是层次分析法，而换乘次数、总时间、花费、步行次数所占的权值大小，由选择路线的人给出。这样基本上覆盖了考虑步行情况的对于每个个体的最优解。

6.3 模型的求解与评价

求解时，我们只给出了步行 1 站的情况，假设 t 等于 10 分钟。

以 S3359 到 S1828 为例，

总时间 64 分钟，换乘 1 次，花费 2 元的路线如下：

$$\mathrm{S3359} \xrightarrow{\text{L015下行}} S2903 \xrightarrow{L485\text{下行}} S1784 \xrightarrow{L167(L217)\text{下行}} S1828$$

经过一次步行 S1784 到 S1828，总时间 66 分钟，换乘 0 次，花费 1 元

7 模型评价与稳定性分析

问题一没有采用传统的图论方法（如 Dijkstra 算法和 Floyd 算法）求解，因为这是有实际意义的题目，单纯考虑图论的经典算法并不能很好的反应这个问题的本质。如果利用 Floyd 算法，实现求解，即求出全局最优解，时间开销是非常大的。因为 Floyd 算法的时间复杂度是 n^3，对于接近 4000 个点的问题，计算量是无法承受的。即使对换乘次数进行了限制，比如在 2 次之内，每求出一对点之间的解预计也需要半小时以上。这样就不具有现实意义。而问题一的算法，是基于公交路线的实际意义给出的，算法更为优化，该模型求解换乘 2 次之内，平均需要不到一分钟（见程序 busrand）。比大多数基于 Floyd 的算法，省时的多，这样就具有了一个查询系统的实际意义。且我们通过程序 busrand 考察出 92%以上的点与点之间是可以在换乘 2 次之内给出可行解。

对于换乘 3 次，或以上的情况，我们根据当前的六对数据可以发现：S3359-S1828 S0087-S3676 的最优解存在于换乘 3 次之内， S1557→S0481 S0971→S0485 S0008→S0073 S0148→S0485 这四对点换乘 3 次对换乘 2 次在时间上的优势并不明显，且花费平均较高。且，基于很多资料给出，成熟的公交系统基本词不需要换乘 3 次以上。因此，我们可以认为对问题 1 的解答是比较完满的。

对问题二的解答延续了问题一的思想，两处核心算法都是基于问题一已实现的程序。一是任意两地铁站求最短路，另一个是公交到地铁的路径。这样我们对问题二的解

答，所需要的时间依然很少，在一分钟之内。在保证问题求解近似最优的情况下，时间上仍然是有优势的。

由于算法求解所需时间很少，因此可以给出任意两点间同时考虑公交网和地铁网的最优路径。那么针对如何评价一套方案，我们给出了如何量化评价的方法。在问题二中我们主要考虑评价方案的三种因素,即换乘次数、时间、价格。在三种因素不冲突(即不存在此消彼长)的情况下，我们用贪心算法的思想要求方案尽量优化每一个因素。这样对于每对起点与终点可给出的方案数大致在 5 套以内，然后利用层次分析法给定相对于每个查询者的最优方案，最终找出与之对应实际的出行路线。

对于**问题**三沿用之前的思想，利用问题而已有的成果，给出了简洁的方法，确定有意义的步行方案。利用问题二中的评价思想，用层次分析法完成对路线的筛选。只是评价因素由三种变成了四种（多加了步行次数）

8 模型扩展 基于 Astar 算法的路径选择问题

由于模型一的算法的初衷，是在限时很短的时间内寻找近似最优解，因此在模型扩展中，我们给出了如何求得全局最优解。而且，基于 Astar 算法，求解所用时间比模型一更少，在毫秒（ms）的数量级。这样，对于公交线路查询的问题，可以认为不论是选择不同评价目标下全局最优的路线，还是实际要求的求解时间上都已达到要求。

Astar 算法是建立在广度优先搜索基础之上的一种启发式算法。在权值不等的情况下，优先考虑生长最优的路径，这样可以极大地优化普通的广度优先搜索。算法复杂度从 n^2 降到了 $\log_2 n$ 。模型扩展中，用 C ＋＋实现 Astar 算法主要利用了堆的数据结构。因为时间有限，模型扩展没有给出合适的估值函数，所以当前的 Astar 算法是只有上段，没有下段的，但这并不影响对此问题的求解。

对问题 1 的验证，发现对站点 S0008→S0073 的最优解出现在换乘次数为 4 时。

“There is a path from 8 to 73 costs 59

The path needs change 4 times busline

S8-S1383-S1691-S3766-S1726-S1008-S940-S2085-S609-S483-S604-S525-S3162-S73”

其它五对站点的答案及问题 1 程序详见附录。求出六对站点路线总耗时 1171MS，即 1.171s

对于问题 2 ，我们比对模型一，给出了更为全面和人性化的解。在对换乘次数、总时间及价钱考虑层次分析时，利用附录第二问 2.txt 中，给出的方案。再具体选择实际中对应的具体路径时，可以参考附录第二问 3.txt 或附录第二问 1.txt，在其中为查询者提供了更多的权值相同路径。这在现实生活中考虑到堵车等现象，更多的路线选择显然有很大的实际意义。即使路线相同，给出更多的可选车次，也可以减少等车的时间。求解所需的耗时，数据最多的附录第二问 3.txt 为例，仅需要 3.2s。

考虑到算法优异的性能，我们对问题 3 也尝试了给出可行的最优解。实现了在模型一提出的对步行 5 站之内不同指标的最优解，这样就可以通过层次分析法，实现对不同需求的查询者最适合的解。

这里给出 S0008→S0073 的解，作为参考，其它解及程序间附录第三问。

We get some best paths from 8 to 73:

There has the shortest path whose length is 54 and costs 8

The path needs change 3 times trafficline

S8-S3412-S2743-S2544-S2953-S778-S2534-D15-D14-D13-D12-D26-D25-S525-S3162-S73

There has the shortest path whose length is 60 and costs 6

The path needs change 3 times trafficline
S8-S3412-S2743-S2544-S2953-S778-P2535-S399-S608-D12-D26-D25-S525-S3162-S73
There has the shortest path whose length is 68 and costs 3
The path needs change 2 times trafficline
S8-S1383-S1691-S3766-S1726-S1008-S940-S2085-P609-S483-S604-S2650-S3470-S2619-S2340-S3162-S2181-S73
There has the shortest path whose length is 77 and costs 2
The path needs change 2 times trafficline
S8-P1381-S1321-S2019-S2017-S1477-S1404-S2480-S3241-S3409-S3186-S3544-S1788-S1789-S1770-S2322-S992-S3142-S3898-S1855-S73

由于时间紧迫，对问题3当前的求解，也许存在小的差错，但已经足以为我们发现趋势，提供一种感性上的参考。

参考文献

[1] 赵巧霞等.以最小换乘次数和站数为目标的公交出行算法.
[2] 杨新苗等.基于GIS 的公交乘客出行路径选择模型[J].东南大学学报(自然科学版),2000,30(6):87 - 91
[3] 韩传峰,胡志伟.城市公交路网性能评估的网络图方法[J].系统工程,2003,21(3):58 - 61.